

Overture

Object-Oriented Tools for Solving CFD and Combustion Problems in Complex Moving Geometry

Technology

Overture is an object-oriented code framework for solving partial differential equations in serial and parallel computing environments. It provides a portable, flexible software development environment for applications that involve the simulation of physical processes in complex moving geometry. It is implemented as a collection of C++ libraries that enable the use of finite difference and finite volume methods at a level that hides the details of the associated data structures, as well as the details of the parallel implementation. While maintaining performance equivalent to hand-coded C or Fortran, Overture is designed for solving problems on a structured grid or a collection of structured grids. In particular, it can use curvilinear grids, adaptive mesh refinement, and the composite overlapping grid method to represent problems involving complex domains with moving components.

The method of adaptive composite overlapping grids provides a computational mechanism to accurately simulate physical processes that are described by systems of partial differential equations (PDEs) in complex moving domains. High-resolution finite difference or finite volume methods are used on a collection of structured curvilinear grids. This basic differencing technology is combined with block-structured adaptive mesh refinement (AMR) to provide local resolution in the computation with correspondingly greater computational efficiency. At Lawrence Livermore National Laboratory (LLNL), this approach is used for high-resolution simulations of incompressible and low Mach number hydrodynamics flows in complex moving geometries. The pay-

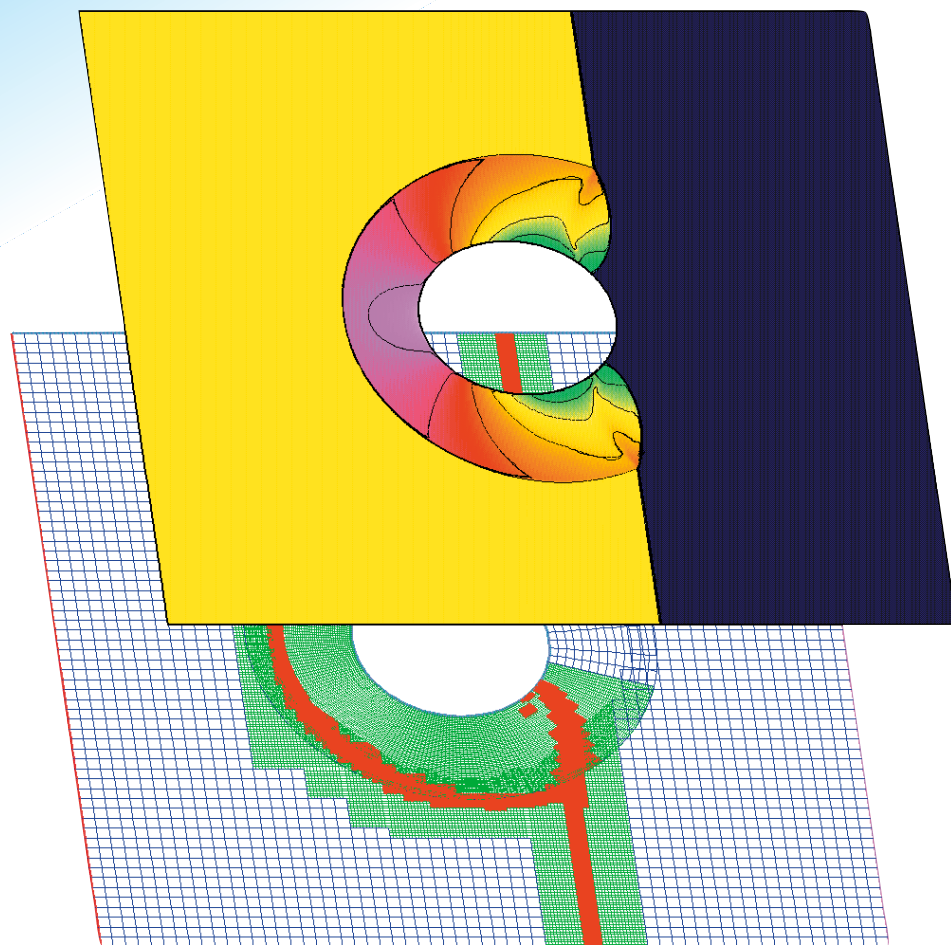


Figure 1. This simulation of a shock interacting with a cylindrical obstacle demonstrates the structured adaptive mesh refinement capabilities within Overture. The two component grids (shown in blue) are refined independently using a Berger-Oliger style approach. The two levels of refinement grids are indicated by the green and red patches.

offs for using this method are significant, but the implementation details can be complex due to the intricacy of both the overall algorithm and the physics behind the simulation.

The Overture Framework

While it is possible to use traditional structured programming approaches to implement, debug, modify, and maintain applications codes based on the adaptive composite overlapping grid method, this would be a daunting task, particularly on parallel computers. The Overture framework was developed using object-oriented design techniques and with the C++ programming language. With object-oriented design, the task is to

develop computational “objects” that represent fundamental abstractions of elements in a computational model. In the structured approach, the fundamental unit of code is a subroutine or function that modifies the data in some way, where in the object-oriented approach, the fundamental unit is an object described by a class in C++. A class contains both data that describe the object and functions that operate on those data.

In Overture, object-oriented design principles have been used throughout to hide the details of complex data structures and algorithms and their parallel implementation. Data structures and algorithms can be specialized and extended through derivation. The

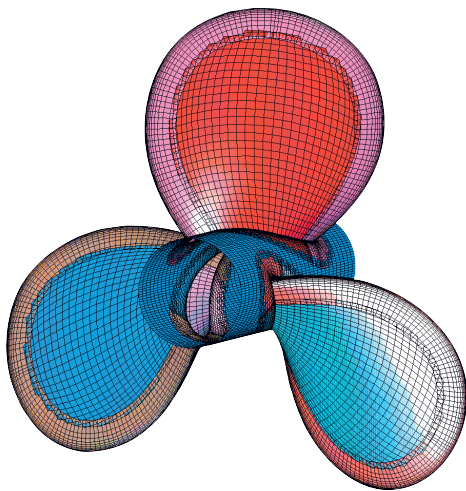


Figure 2. Overlapping grids can be used to represent complex geometries by combining component grids. This example uses component meshes provided by engineers at Chalmers University of Technology, Sweden.

Overture classes provide tools for the rapid development of application codes. The main class categories are listed below.

1. A++/P++ arrays describe multidimensional arrays and provide for serial and parallel operations on those arrays. In the parallel environment, these provide for the distribution and interpretation of communication required for the data-parallel execution of operations on the arrays.
2. Mappings define transformations such as curves, surfaces, areas, and volumes. These are used to represent the geometry of the computational domain.
3. Grids define a discrete representation of a mapping or mappings. These include single grids and collections of grids, in particular composite overlapping grids. CASC's Rapsodi project provides tools for the construction of curvilinear grids and for overlapping those grids to represent complex moving geometries.
4. Grid functions provide for the representation and centering of solution values such as density, velocity, and pressure, defined at each point on the grid(s).
5. Operators provide discrete representations of differential operators and boundary conditions through finite

difference or finite volume approximations.

6. Visualization tools based on OpenGL are provided to furnish a high-level graphics interface for visualizing geometry and simulation results.
7. Adaptive mesh refinement provides automatic refinement of the overlapping grid structure for increased local resolution and efficiency of computational simulations.
8. Load-balancing tools are provided for automatic load-balancing of computations on the adaptive overlapping grid structure on parallel computers.
9. Parallel distribution mechanisms are provided through the PADRE library, part of the DOE 2000 ACTS toolkit.
10. Full Fortran-like performance provided by the ROSE optimizing source-to-source code preprocessor.

Overture Application Code Development

Scientists at LLNL collaborate with academia to use Overture to develop flow solvers for high-speed compress-

ible flow problems, incompressible flow problems, low Mach number, non-Newtonian, and reacting fluid flow problems.

Through its object-oriented design, Overture reduces code duplication, encourages interoperability of application software, and simplifies the learning curve for new computational methods. Overture's object-oriented architecture provides flexibility to address a wide range of applications that involve simulations in complex moving geometry on serial and parallel computers. The advantages of this approach include reduced code development time and broader, more in-depth research into numerical methods for scientific and industrial applications.

For additional information about Overture, contact David L. Brown, (925) 424-3557, dlb@llnl.gov; or William Henshaw, (925) 423-2697, henshaw@llnl.gov; or Daniel J. Quinlan, (925) 423-2668, dquinlan@llnl.gov. <http://www.llnl.gov/casc/Overture/>

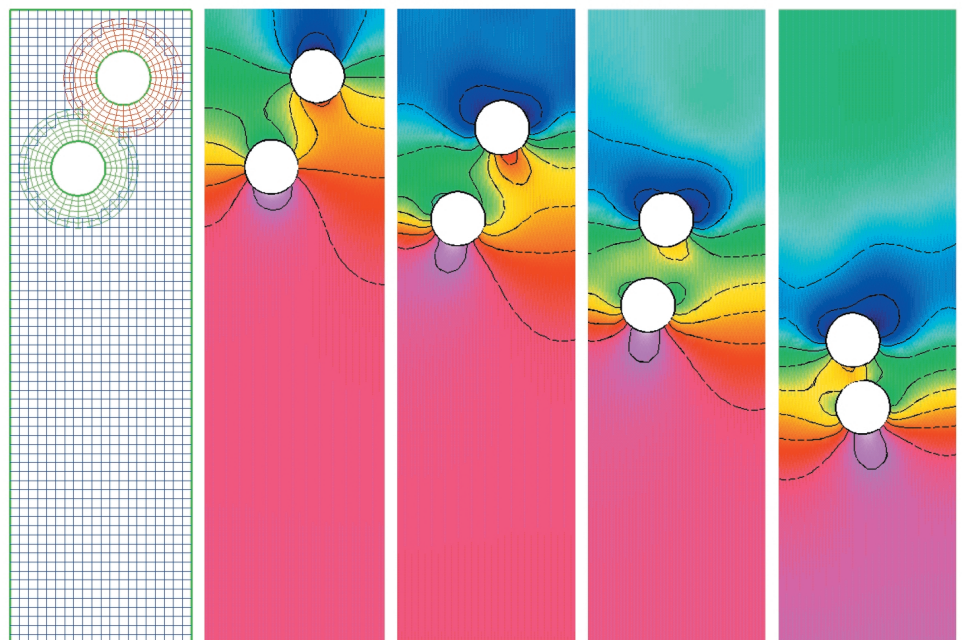


Figure 3. The moving geometry capabilities of Overture are demonstrated in this figure. Here two solid drops interact with a viscous fluid under the force of gravity. The use of overlapping grids is an efficient approach for computing such flows since the grids do not have to be regenerated as the solution evolves. Only the locations of the interpolation points used to communicate information between the grids is recomputed at each time step, a much more efficient operation.